Image Thresholding and Blob Tracking

2/12/17

Josh Powell
Yosuke Higashi

Figures

Figure 1. Before morphological operator image 1

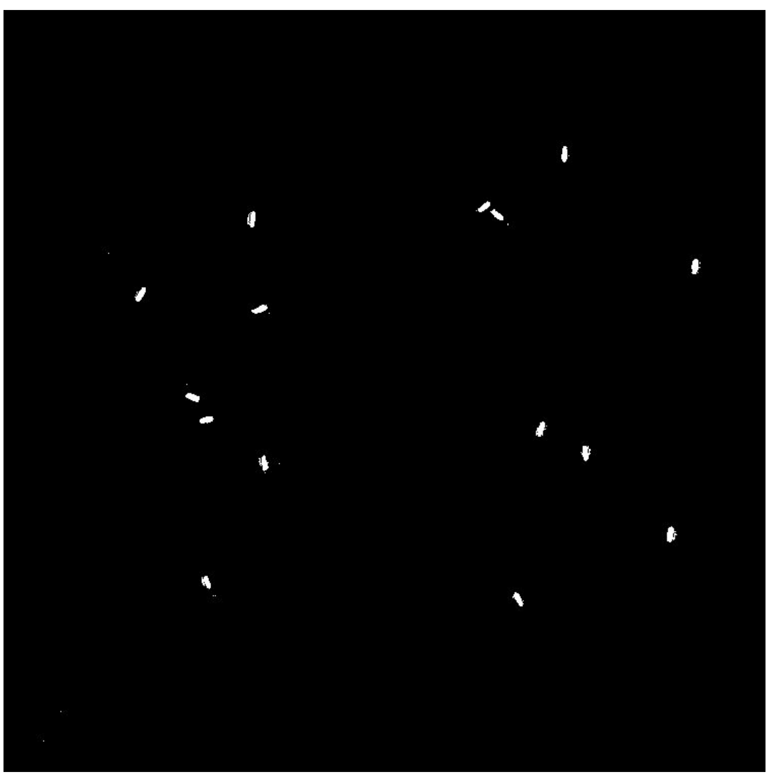

Figure 2. Before morphological operator image 2

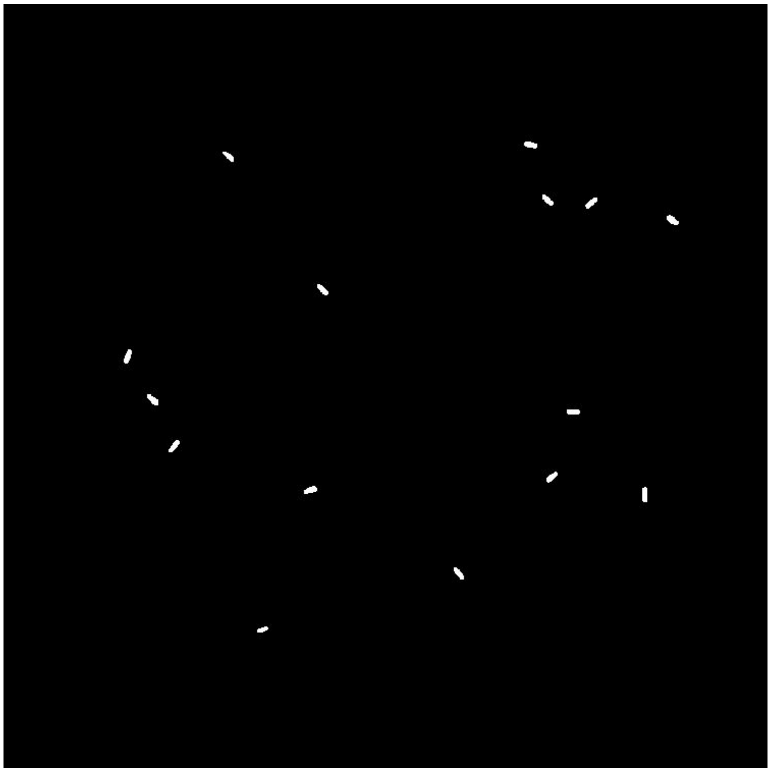

Figure 3. After morphological operator image 1

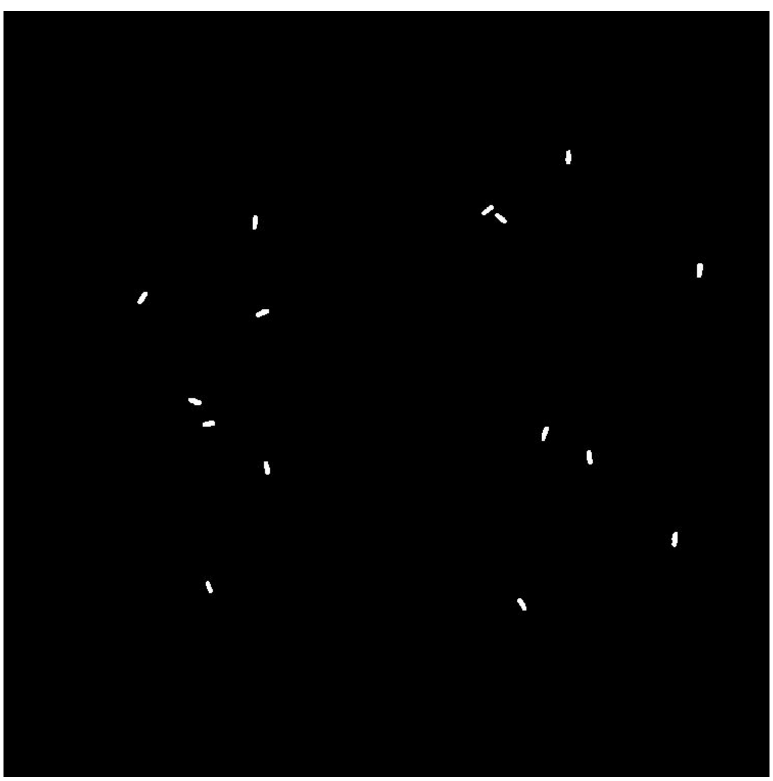

Figure 4. After morphological operator image 2

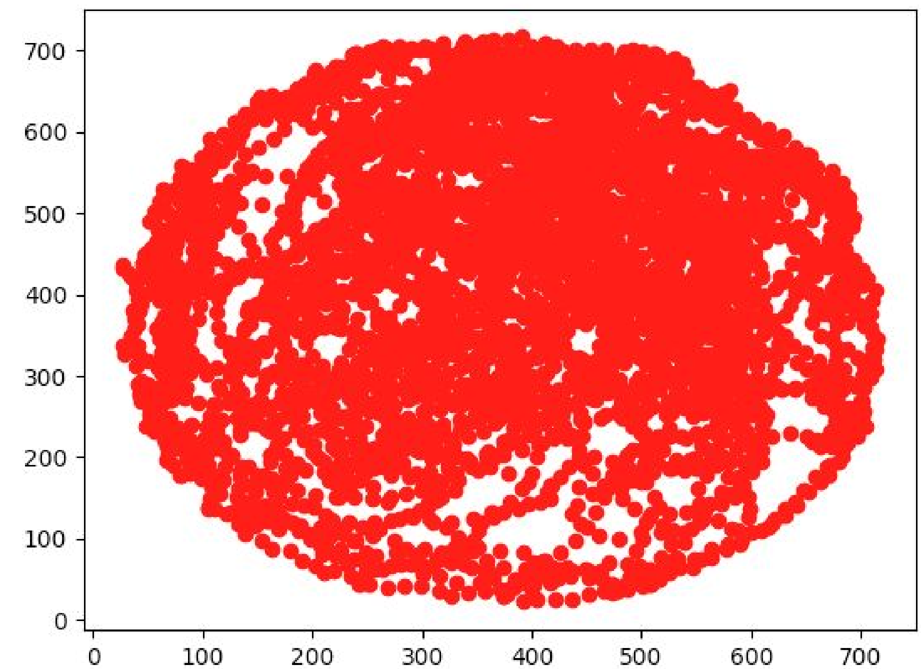

Figure 4. Single (x,y) plot of disconnected points showing the positions of the centroids of the connected components in the thresholded images

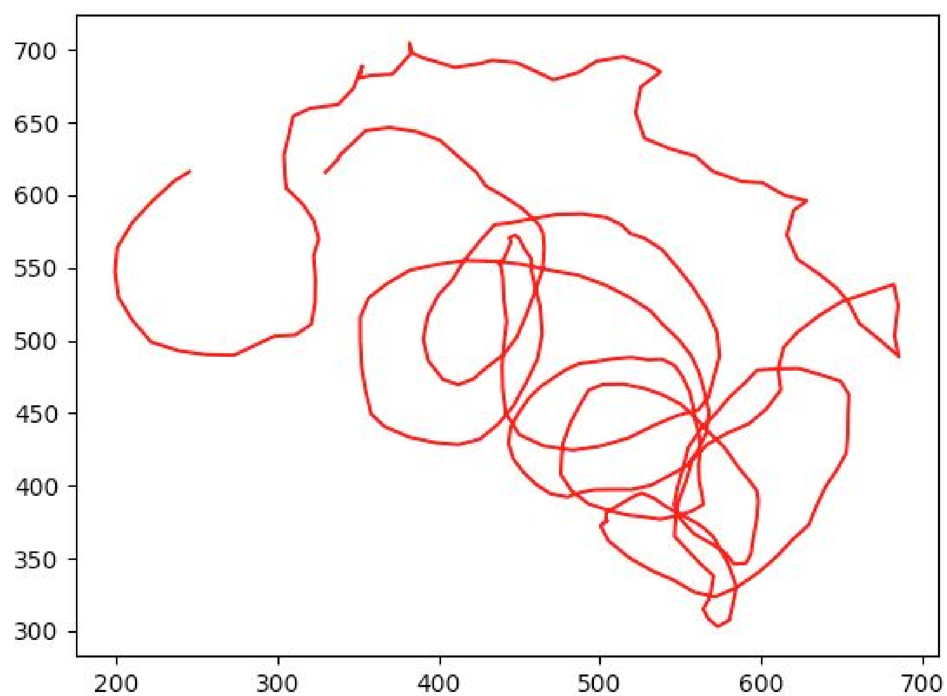

Figure 5. Plot of connected points showing the trajectory of fly 0

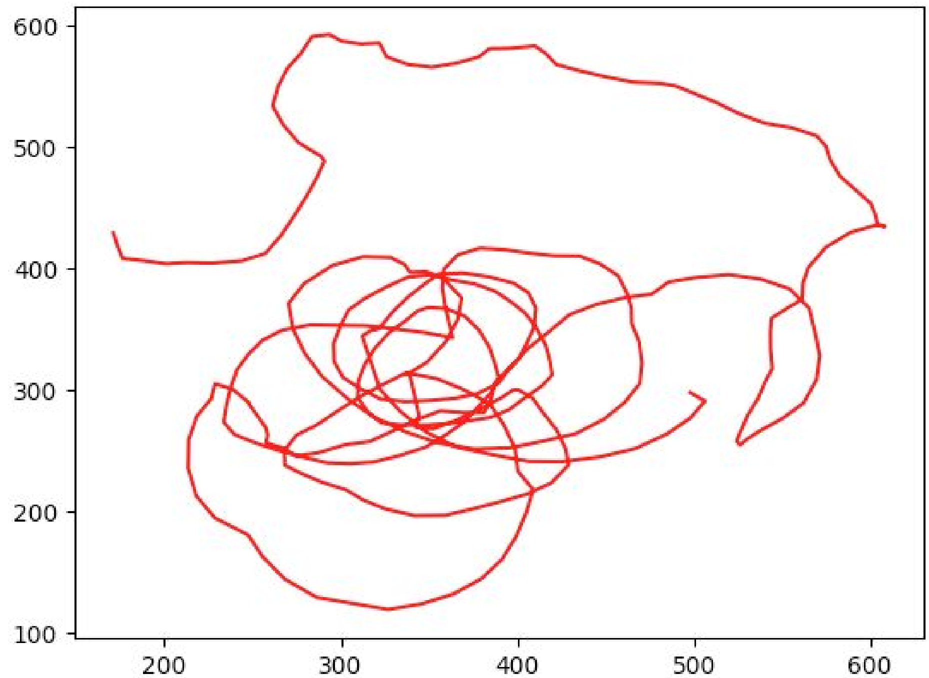

Figure 6. Plot of connected points showing the trajectory of fly 5

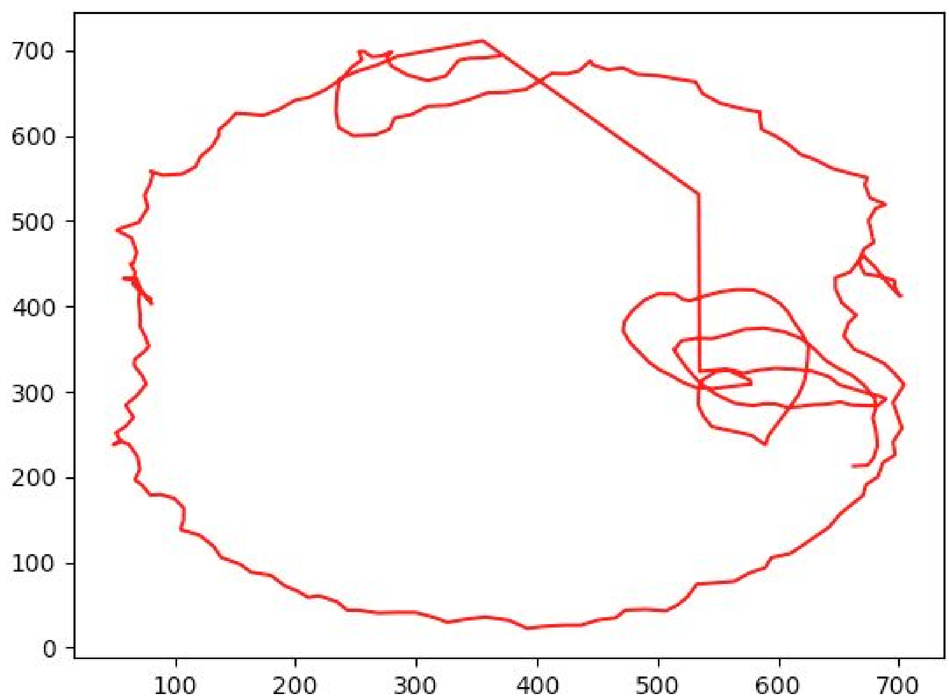

Figure 7. Plot of connected points showing the trajectory of fly 10

Discussion

Thresholding and Morphological Operators

In order to do background subtraction, a background was defined. The pixel intensity of the first five hundred frames were averaged. Although there are flies in each frame, since the pixel intensity is averaged out for a large number of frames, the resulting image is only the empty dish. The video is read and the background image is subtracted from each frame. This background subtraction yields frames of the flies with a black background. The flies appear to be white but the background subtraction does not give a binary image. Therefore a binary threshold is necessary. As a result of the binary threshold, each frame has white flies with a black background and some white speckles. The proper morphological operator for eliminating white speckles is opening. It was implemented by using the morphologyEx function.

Connected Components Analysis

The findContours function gives outlines of connected components of non-zero pixels. In this case, it returns the outlines of the flies. The mean of the outline pixels for each fly was computed. This is the centroid of each fly. From the connected component analysis, outlines of each fly can be applied to each frame. In addition, a list of the centroids of each fly in each frame can be determined. This list is not ordered and further analysis will be needed in order to track individual flies.

Tracking

The goal of tracking is to take an unordered list of centroids that are produced at each frame and figure out which fly each centroid corresponds to. The general approach used was creating a circular mask, or area of interest, in front of each fly. The center of the mask is a distance in front of the current position of the fly equal to the previous translation vector. In other words, if the fly moves from frame 1 to frame 2. A translation vector [x,y] is determined. This translation vector is used to determine the center of the area of interest in frame 2 in order to estimate the fly's position in frame 3. Suppose there are multiple flies in a given fly's area of interest. Some safety code was implemented and the fly closest to the center of the area of interest was determined to be the given fly's next position. One consideration of the area of interest is if the fly makes a sharp turn. Since areas of interest were produced in each frame, even if the flies move quickly or take sharp turns, they are still encapsulated by the area of interest.

Two dictionaries were used. One maps a fly number to the centroid and the other maps the fly number to the translation vector. Starting with the disordered list of centroids, the translation dictionary, which stores the translation vectors of the previous frame to the current frame, is used to make the area of interest for each fly. In the current frame, say there is only one fly in the next frame in fly 1's area of interest. This fly is fly 1's next position. If there are multiple flies in fly 1's area of interest, the fly closest to the center of the area of interest is fly 1's next position. The centroid dictionary is updated with this data. In the code, the variables centroids_used and centroids_unused keep track of which centroids have been added to the centroid dictionary and which have not. The code ensures that each centroid gets paired with the most likely fly, no fly gets paired with multiple centroids, and no centroid gets paired with multiple flies.

Centroids and the fly numbers they describe were printed. This data was used in the code in the data visualization in figures 4, 5, 6, and 7. The centroids for each fly in each frame were plotted in figure 4. Additionally, figures 5, 6, and 7 are plots of the trajectory of individual flies as shown.

As shown by these figures, the code works very well as the trajectories of individual flies are continuous curves. The area of interest approach prevents swapping of fly numbers from occurring and safety code prevents flies getting assigned multiple centroids or no centroids.

Interactive Features

When starting the program, the user has an option of either viewing the binary video or the normal video. The normal video shows the original video with a white circle around the flies and a number above each fly which corresponds to the fly number. The binary video simply shows the flies as white and the background as black. If the user pushes the “+” key or “=” key, the threshold value is increased, and if the user pushes the “-” key, the threshold value is decreased. This way, the user can adjust the threshold value in real time and can also see the effect of increasing or decreasing the threshold value.